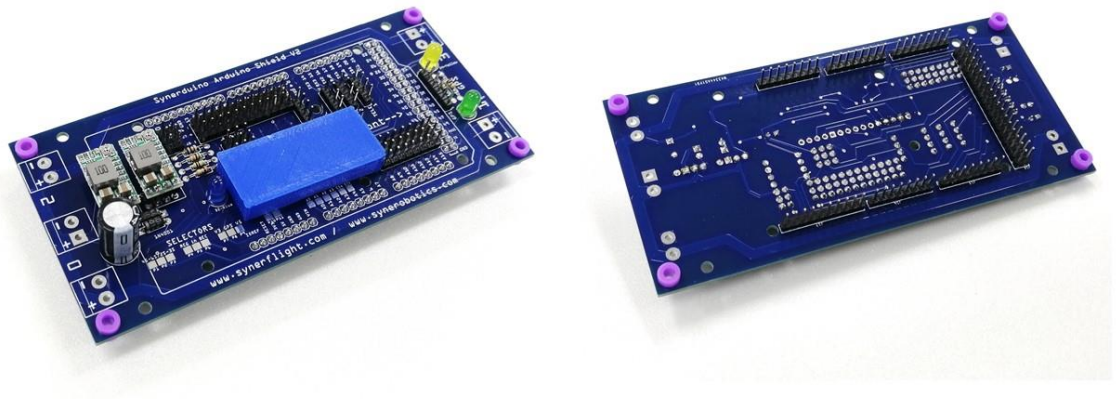
SYNERFLIGHT



Synerduino Arduino shield 2560 v2

This Documentation for Those in the Philippines who wish to use something Locally made we have a special Documentation just for the Synerduino Expansion option.

Support for Synerduino Arduino Board Classics
1 I2C GPS
1 UART
Light weight Mission planning
Basic GPS Mode
Navigation Max of 15WP

e-Gizmo MECHATRONIX CENTRAL

GizDuino UNO SE





GizDuino

GizDuino LIN



SYNERDUINO BOARD PREPARATION

Aux ADC in

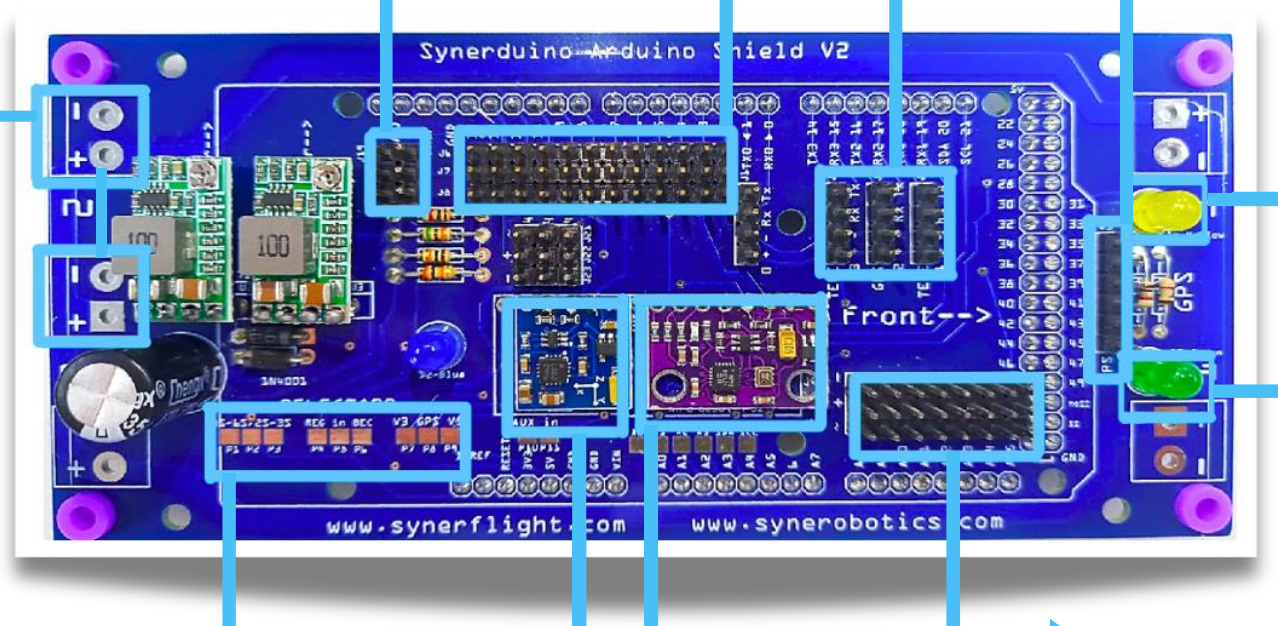
Usage: These pins are used to connect external sensors or devices that output analog signals, such as temperature, pressure, or current sensors.

ESC / Servo PWM Out

Usage: ESCs should be connected to these pins for motor control in a drone or robot.

Serial Pins

Usage: You can use these for connecting devices that need to send/receive serial data.



GPS Serial Pins

Usage: You will connect the GPS module's TX to the RX pin on the board and vice versa, ensuring the correct serial data exchange between the GPS and the microcontroller.

Power Input

Usage: Connect a 3S LiPo battery to this input. Make sure to match the polarity of the battery with the input terminals to avoid damaging the board.

Soldering Note: ESCs should only be soldered on the top side of the board, ensuring the solder joints do not penetrate through to the bottom.

GPS LED

Usage: When the GPS module is properly connected, this LED will inform you of its status. A blinking or steady light usually indicates good GPS reception.

Status LED

Usage: The microcontroller on the board controls this LED, which will light up to signal that the board is functioning correctly.

Jumper Pads Selector Zone

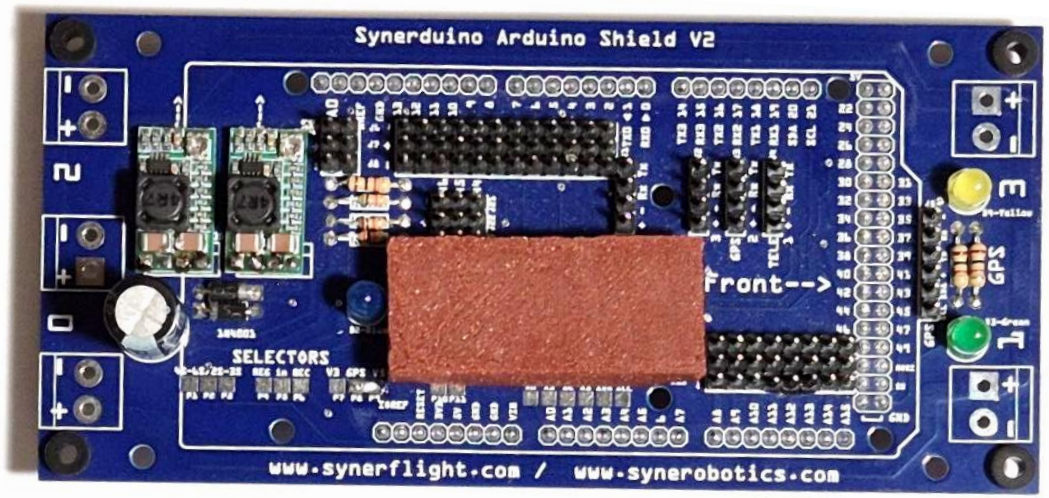
Usage: By soldering the appropriate jumper pads, you can select between different power sources or set the board for the correct number of battery cells.

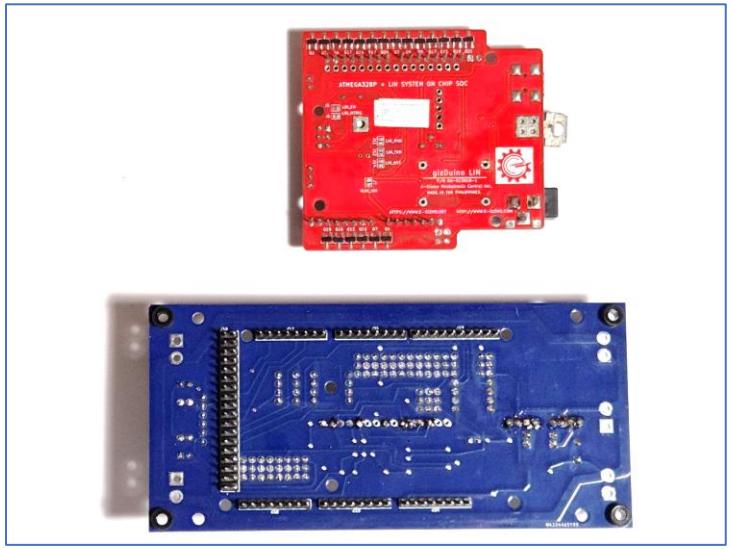
BMP180 – 85 Baro

IMU : MPU-9250

RC PWM In

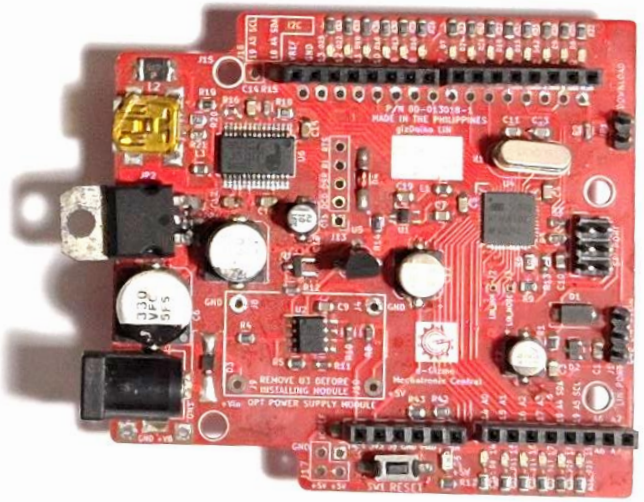
Usage: Connect the output pins from your RC receiver to these PWM input pins. The signals will then be processed by the microcontroller, which can send commands to the motors through the ESC.

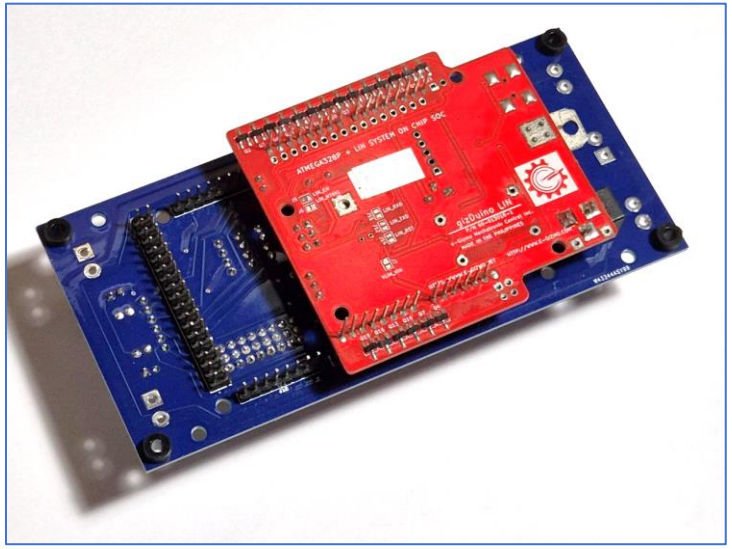




As Simple as attaching any other Arduino UNO Board

In this sample we use the Gizduino LIN

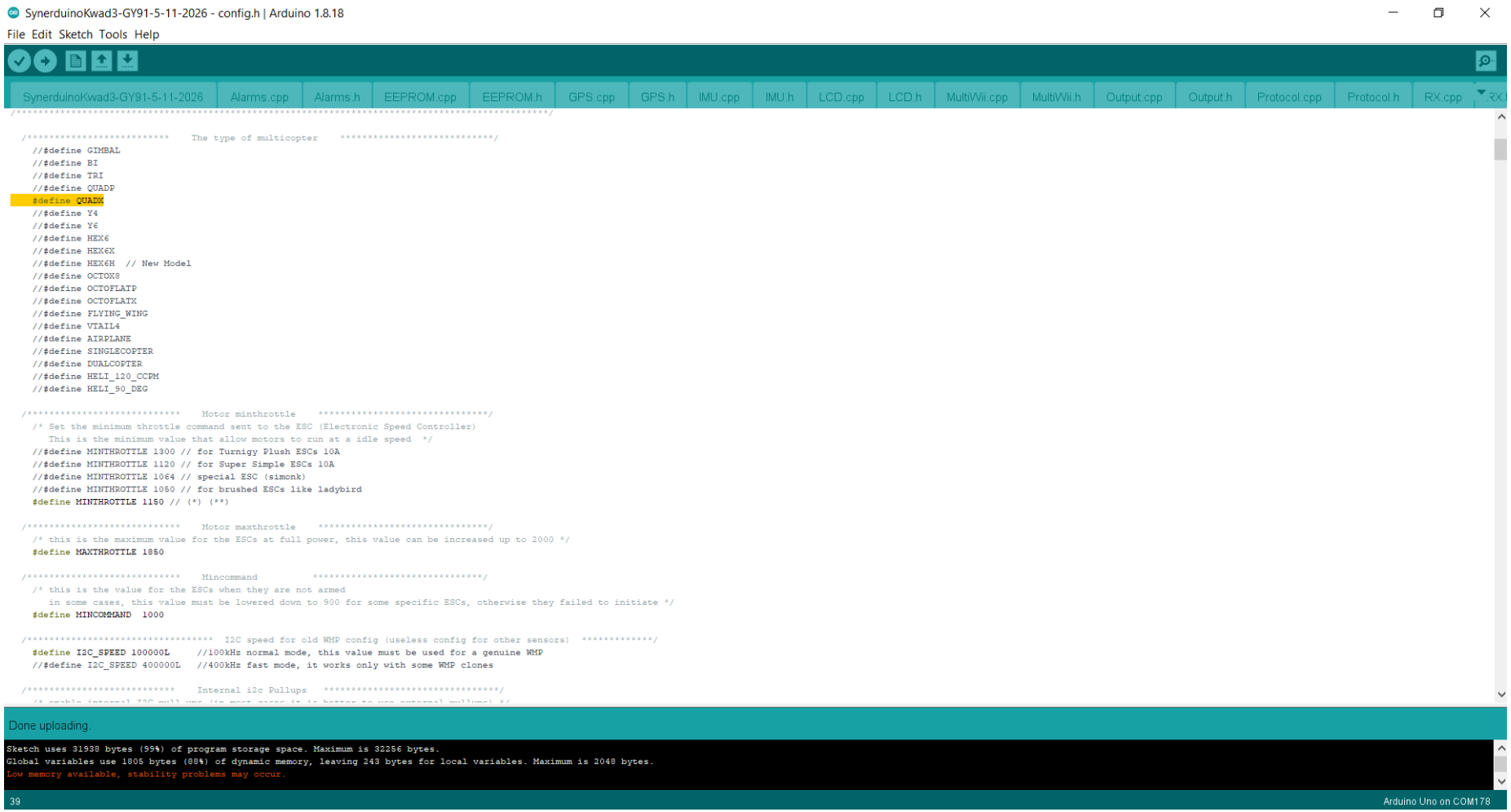




Open your Synerduino Ino

In Config.h
Select your parameters base on your vehicle platform and serial setup till you get to the GPS setting part

Synerduino INO can be downloaded thru the Synerduino Arduino classic product page



Select your Synerduino Board model



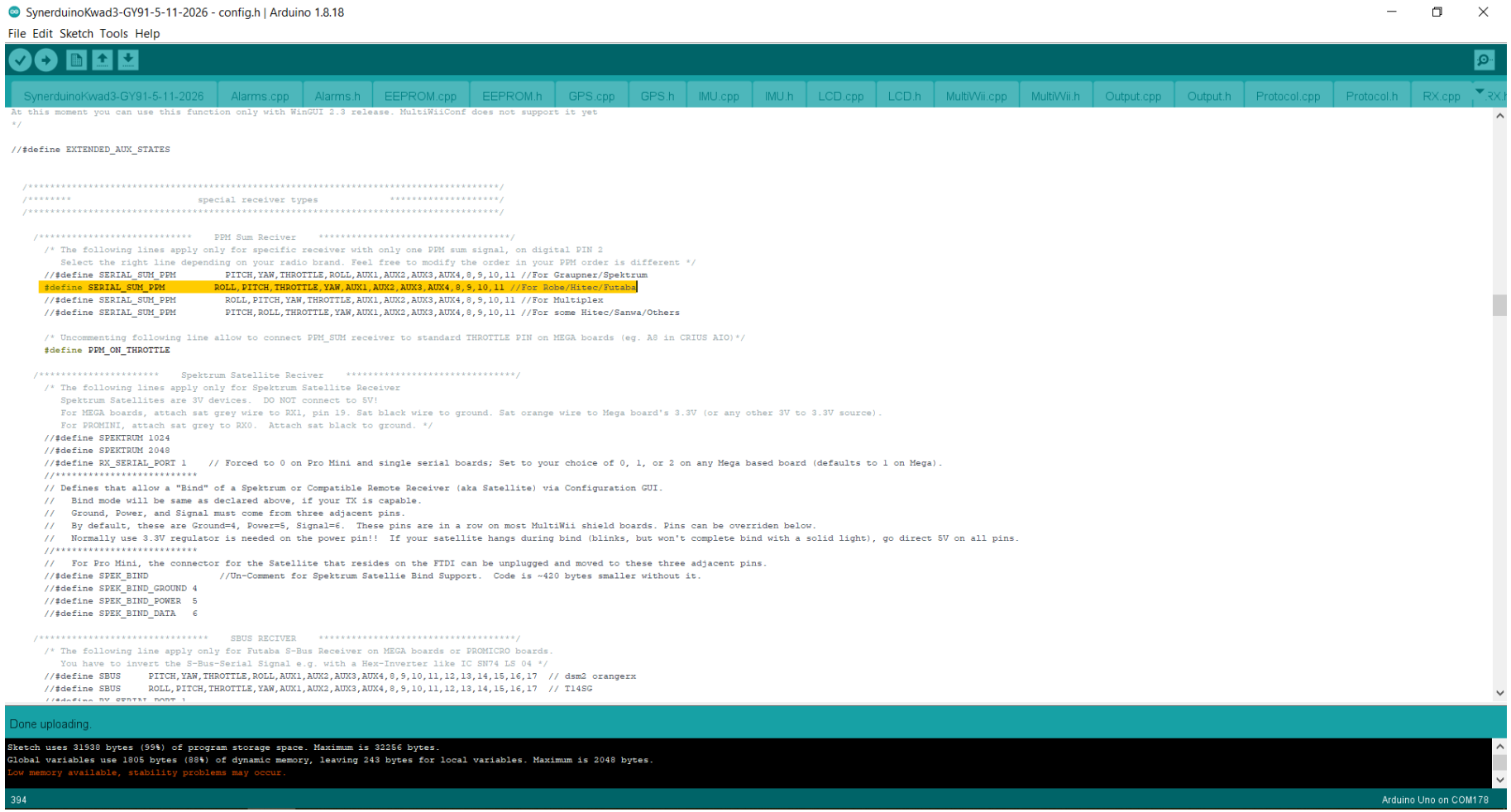
Special Receiver Type
By Default when both PPM and Sbus is commented

its set as PWM using A8 to A15

To save space we can
Use A8 as PPM if your receiver supports

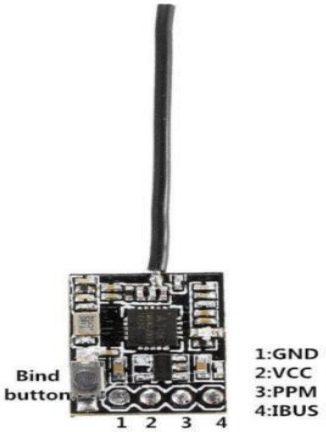
Uncomment
#define PPM_ON_THROTTLE

#define SERIAL_SUM_PPM
(the receiver layout model you have)



RECEIVER TYPES PrOTOCOL





PPM RECEIVER





SERIALRECEIVER wont be used here as it would occupy the space reserve for Telemetry







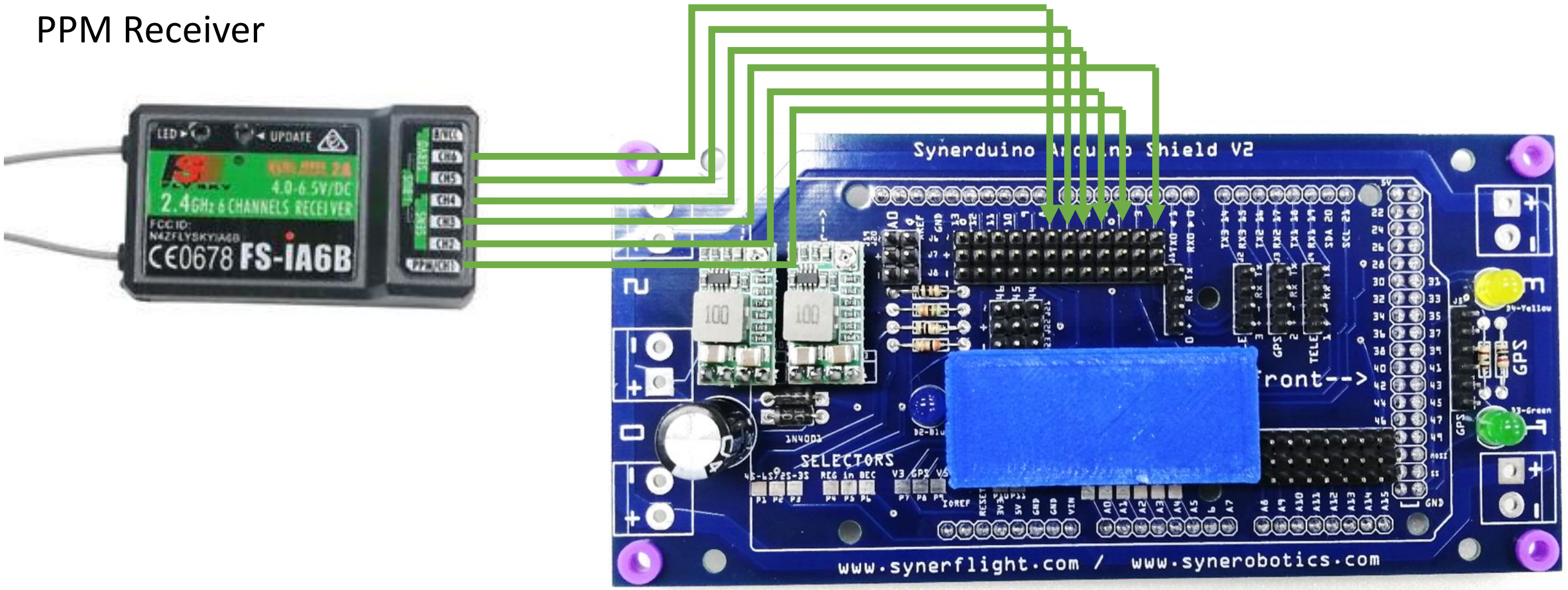
PWM RECEIVER

RECEIVER TYPE CONFIGURATIONS

RX > SBUS INPUT	FUTABA FORMAT	JR FORMAT	WALKERA FORMAT	GRAUPNER FORMAT	UNO
	AETR	TAER	EATR	ERTA	INPUT
					
THROTTLE	CH3	CH1	CH3	CH3	D2
AILERON	CH1	CH2	CH2	CH4	D4
ELEVATOR	CH2	CH3	CH1	CH1	D5
RUDDER	CH4	CH4	CH4	CH2	D6
AUX 1	CH5	CH5	CH5	CH5	D7
AUX 2	CH6	CH6	CH6	CH6	D8
AUX 3	CH7	CH7	CH7	CH7	N/A
AUX 4	CH8	CH8	CH8	CH8	N/A

OTHER RECEIVER TYPE

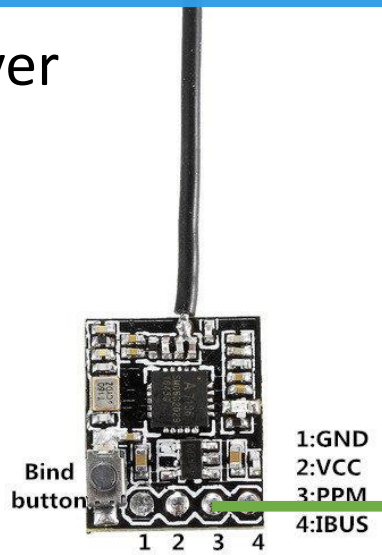
PPM Receiver



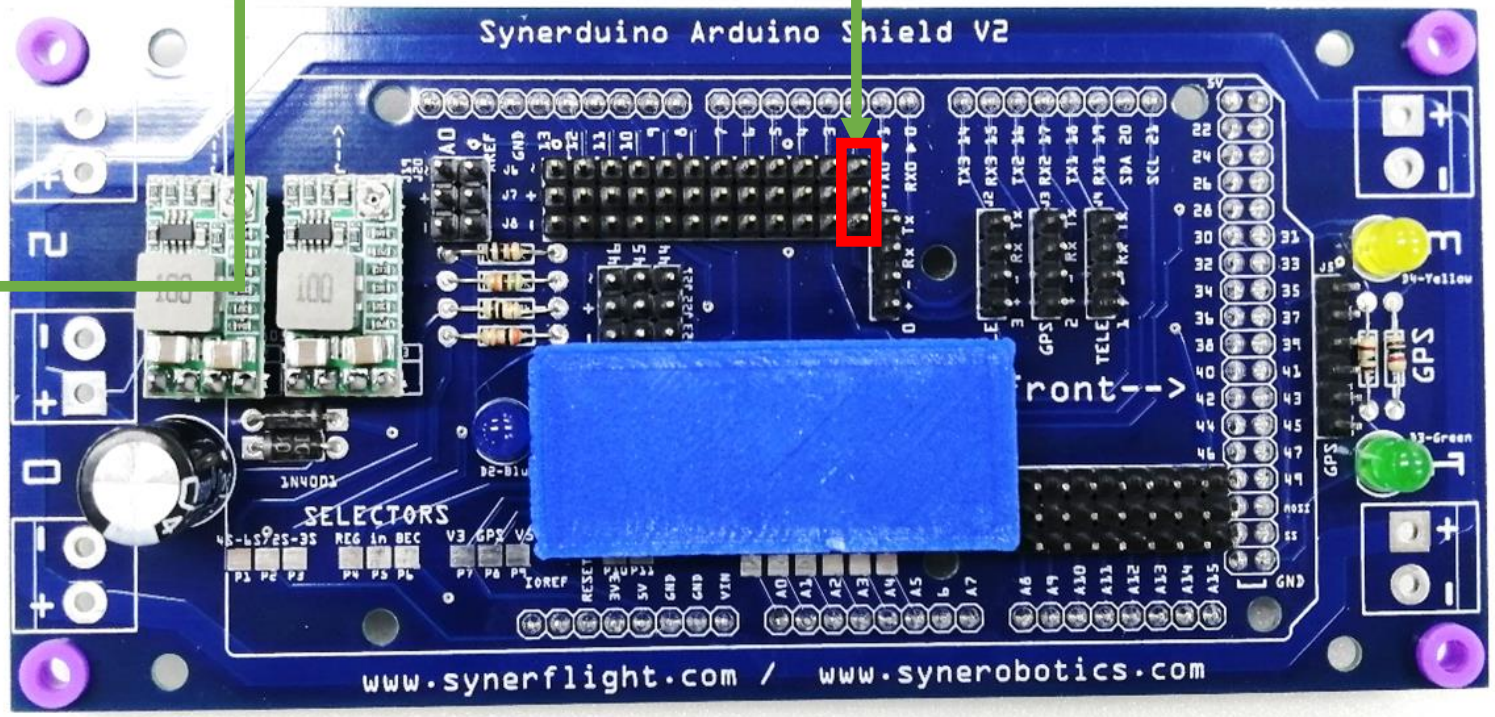
Pin D2 for Uno

OTHER RECEIVER TYPE

PPM Receiver



Pin D2 for Uno



TELEMETRY COM SPEED

CONFIG.H

SERIAL BAUD RATE

WILL DEPEND ON WHAT BAUD YOUR TELEMETRY MODULE ARE SET INTO YOU CAN CHANGE IT TO SUITE THE PORT YOUR DEVICE IS CONNECTED TO.

SERIAL 0 CAN BE USE FOR TELEMETRY GIVEN NOTHING IS CONNECTED TO THE USB AT THIS POINT. AND FIRMWARE MUST BE FLISH PRIOR TO HOOKING UP ANYTHING TO THIS PINS

SERIAL 1 , 3 IS RESERVE FOR TELEMETRY

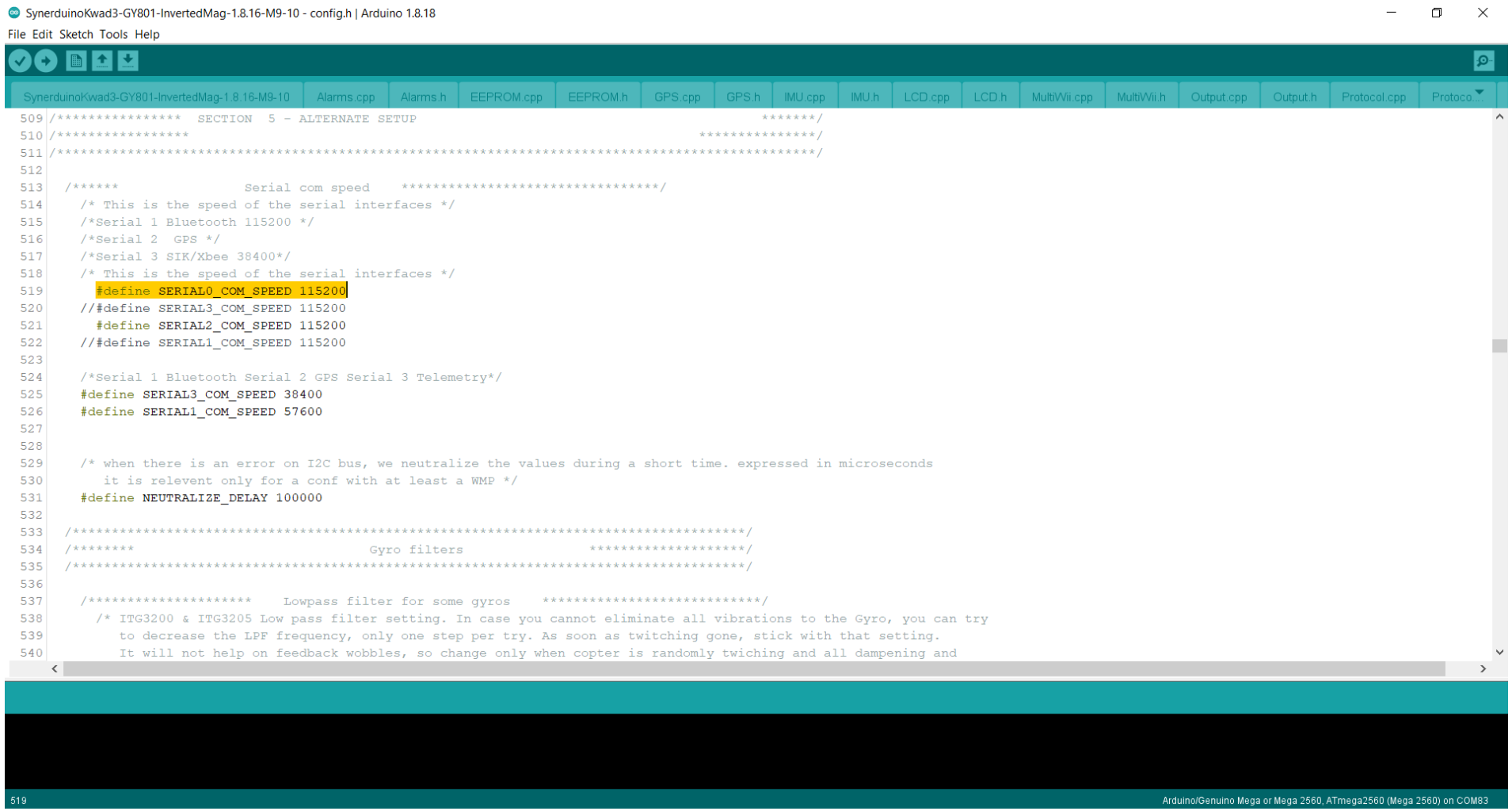
115200 FOR BLUETOOTH HC-05

38400 FOR XBEE RADIO

57600 for SIK RADIO

SERIAL 2 IS RESERVE FOR GPS

NMEA Baud 57600



TELEMETRY



38400 OR 57600 FOR SIK RADIO DEPENDING IF USES 433MHZ OR 900MHZ (63kbps)



38400 FOR XBEE RADIO



115200 FOR BLUETOOTH HC-05

STANDARD FOR ALL DRONES TO USE SERIAL LINK AS TELEMETRY MAINLY ON YOUR TX RX SERIAL PORTS , NOTE: THE LOWER THE FREQUENCY OF THE RADIO THE LOWER THE BAUD IS NEEDED ,

MOST DRONES REQUIRE MINIMUM 63kbps AIRSPEED TO COMMUNICATE PROPERLY

PROTOCOL IS MSP RAW OR MAVLINK

BLUETOOTH CONFIGURATION

TO CONNECT THE **BLUETOOTH HC-05 MODULE** TO THE BOARD, ENSURE THE HEADERS ARE ARRANGED CORRECTLY. FOLLOW THIS WIRING CONFIGURATION:

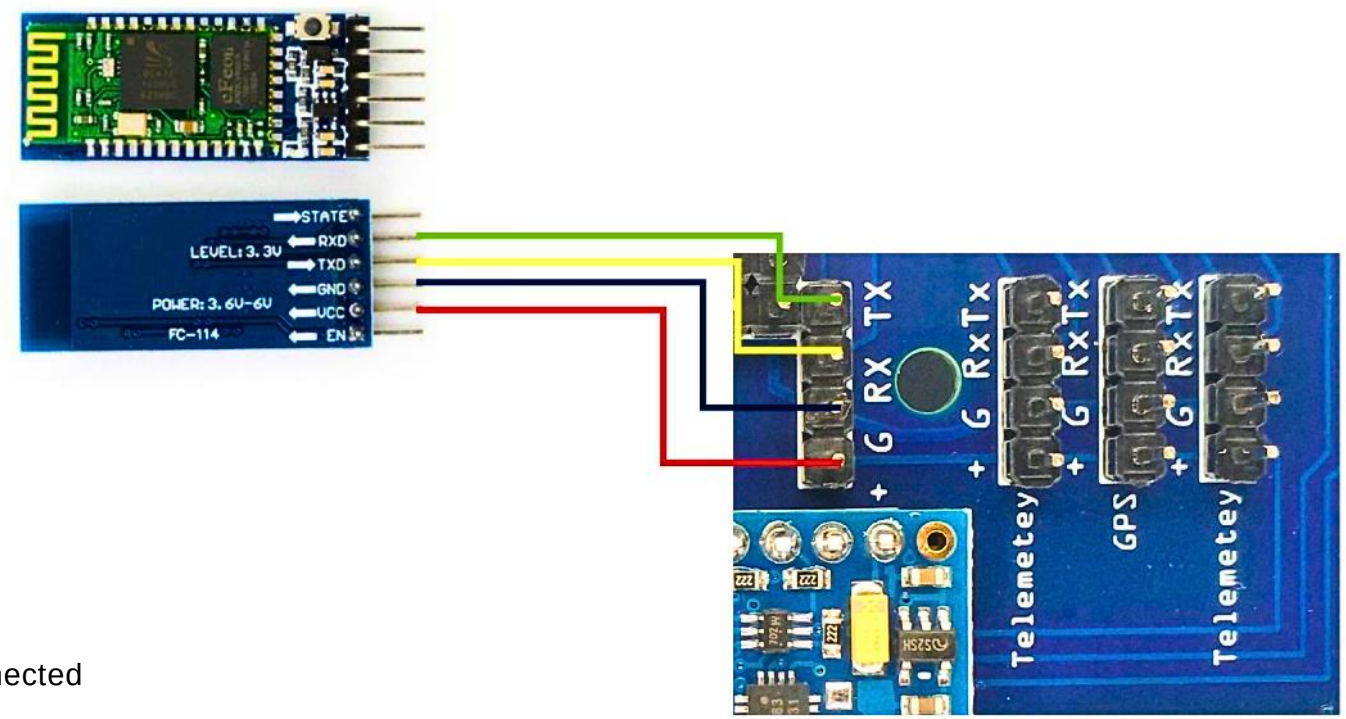
Because the Guizduino only has one serial its
Obviously to be UART0

VCC (Bluetooth) connects to **+** (Board)

GND (Bluetooth) connects to **G** (Board)

TX (Bluetooth) connects to **RX** (Board)

RX (Bluetooth) connects to **TX** (Board)



Any Serial Radio can be configured to run on
Serial 0, 1, or Serial 3. with the matching Baud

Serial 0 can be used for telemetry only if the USB is disconnected

NOTE: DOUBLE-CHECK THAT THE WIRE COLORS MATCH THESE MARKINGS. INCORRECT INSTALLATION OR POLARITY MAY DAMAGE THE ARDUINO BOARD. THE BLUETOOTH MODULE IS PRESET TO A BAUD RATE OF 115200 FOR YOUR CONVENIENCE, BUT YOU MAY CHANGE THE SETTINGS IF NEEDED.

On GPS

Should you choose to use a GPS you would need to use an Serial –I2C converter

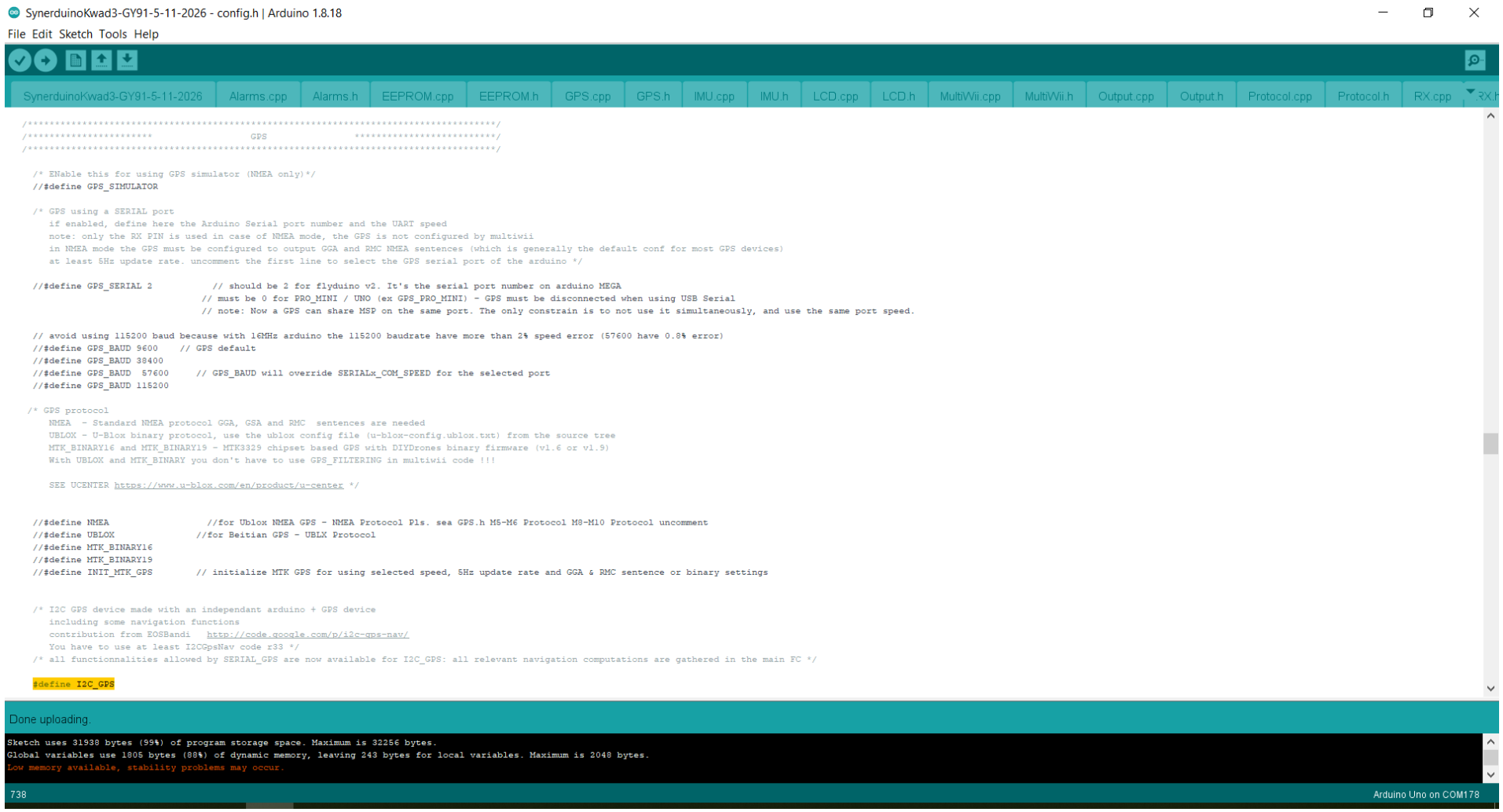
By Default
Serial

#define GPS_SERIAL 2
#define GPS_BAUD 57600
#define UBLOX

These are to be commented as we are using the I2C for our GPS

uncomment
#define I2C_GPS

Comment the Sonar when when you have Sonar installed on your 2nd board
#define I2C_GPS_SONAR



I2C GPS NAV

Synerduino I2C GPS

this allows smaller 328P & ATA66140 Boards to utilizes I2C GPS converter instead of directly hooking up the GPS to the Main UART due to its limited Size and memory.

Synerduino I2C GPS NAV

Synerduino Multirotor Ino

pls see documentation and its associated sensors inorder to download the version that matches your board, this is Arduino IDE Versions of the said firmware

Notes:

http://synerflight.com/wp-content/uploads/2026/11/i2c-gps-nav-master-2026.zip

For those who wish to add GPS function the latest firmware now supports M8-M10 GPS modules while the older Firmware supports M5-M7 GPS Modules

Because Gizduino only has 1 Serial being UART 0 the GPS has to go somewhere so download this from the Download tab and upload it to a end Gizduino board

This Firmware runs a serial I2C conversion for Multiwii allowing it to be attach to the SDA and SCL Pins of your Gizduino

The setting

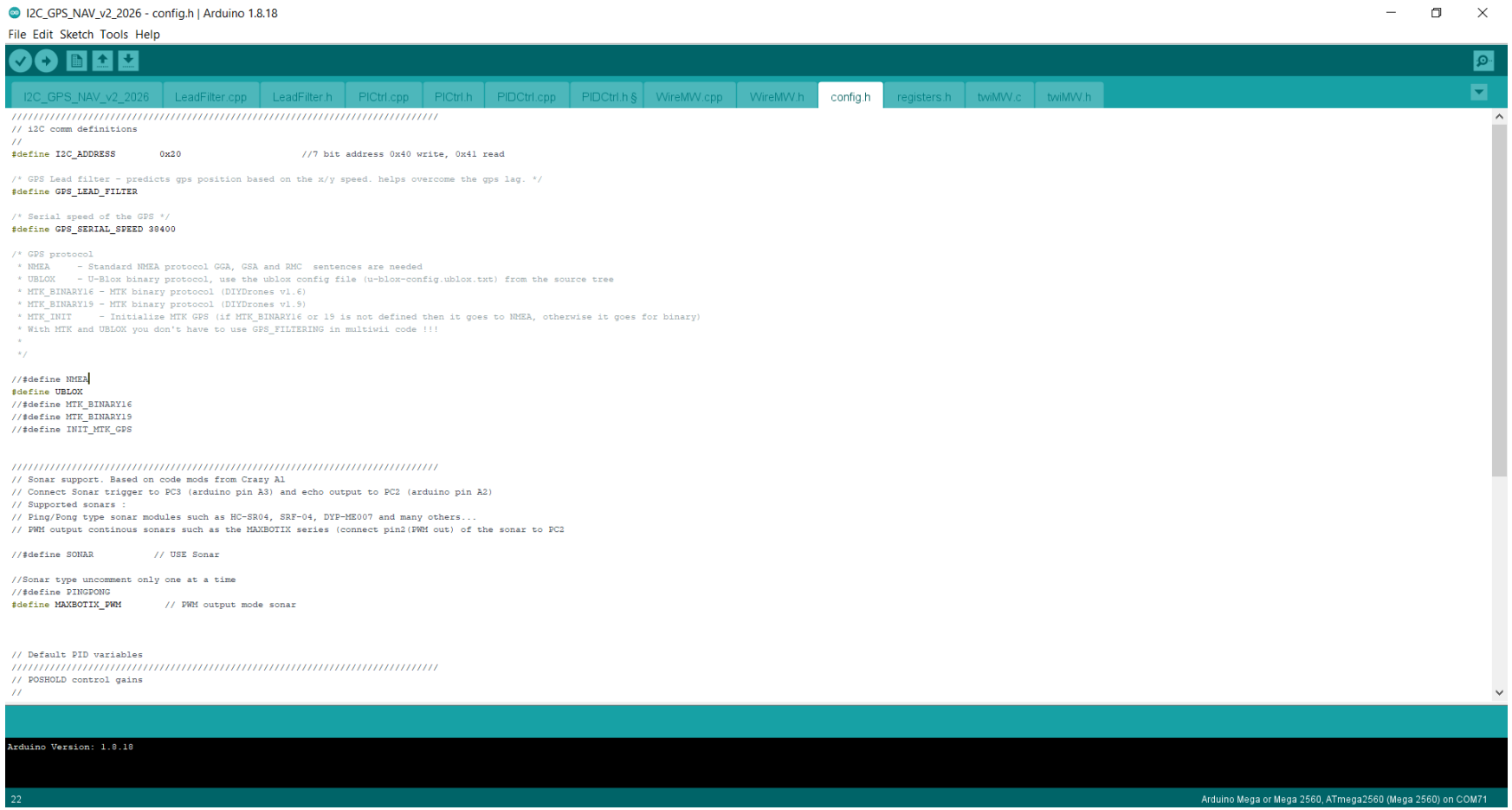
/* Serial speed of the GPS */
#define GPS_SERIAL_SPEED 38400

#define UBLOX

If you need sonar
//#define SONAR // USE Sonar

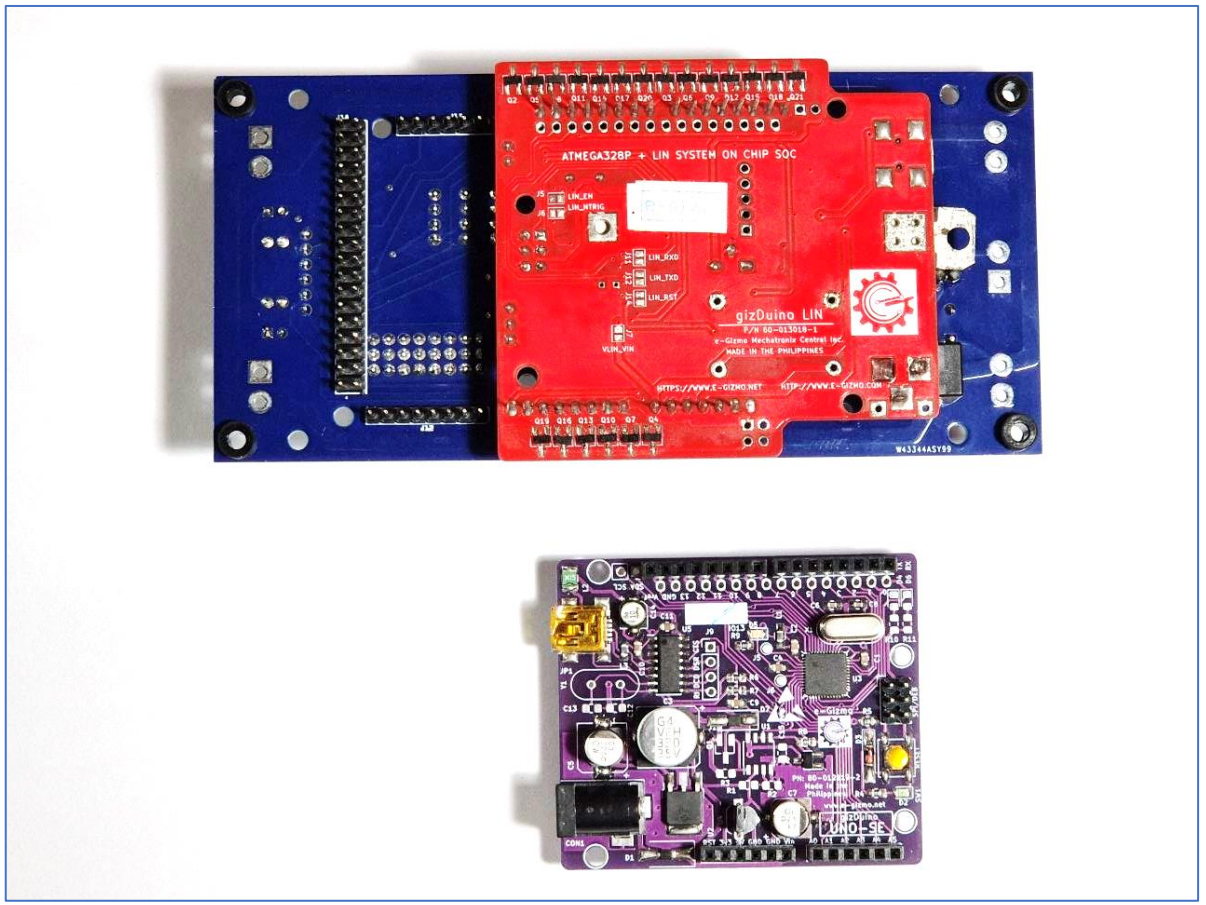
Connect Sonar trigger to PC3 (arduino pin A3) and echo output to PC2 (arduino pin A2)
// Supported sonars :
// Ping/Pong type sonar modules such as HC-SR04, SRF-04, DYP-ME007

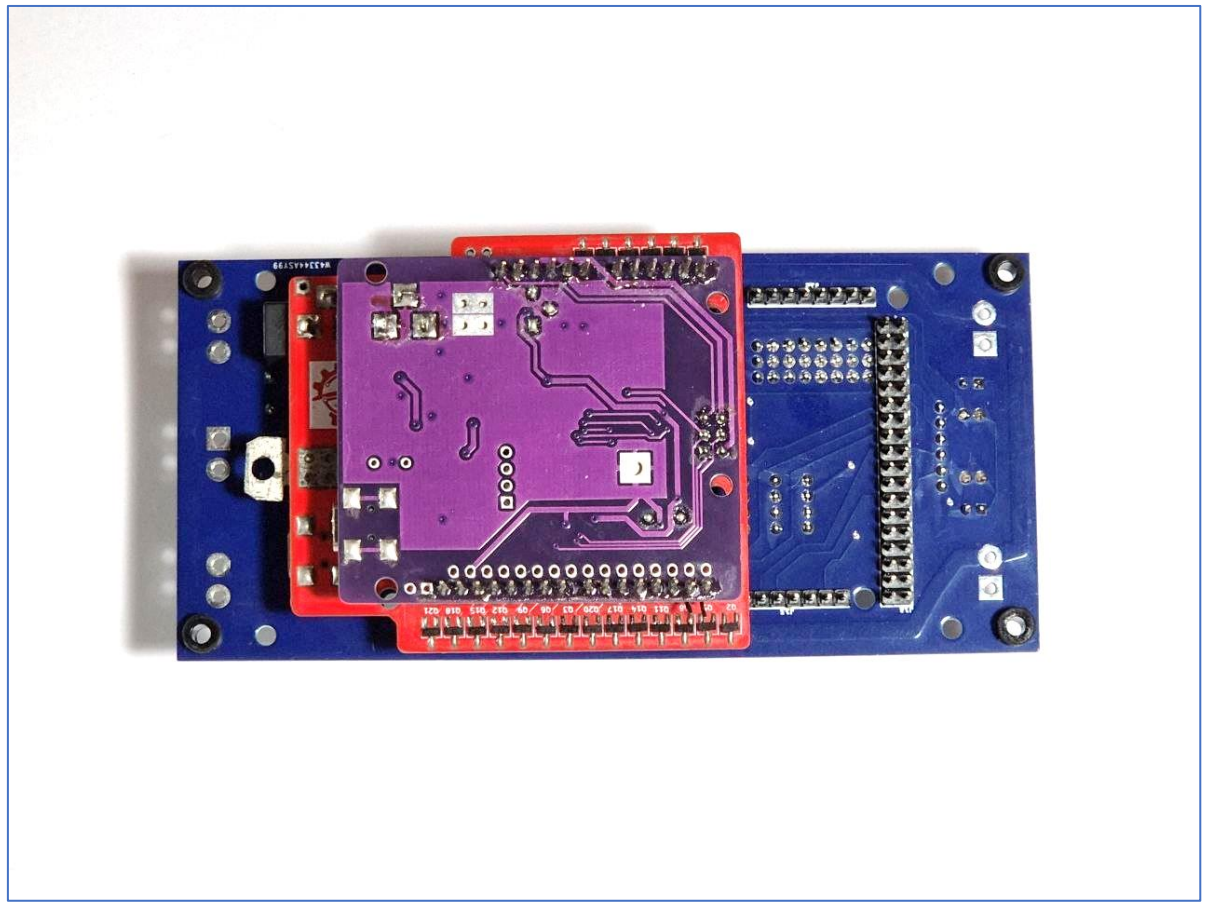
//Sonar type uncomment only one at a time
//#define PINGPONG



Another Guizduino board would serve as the Serial to i2C GPS converter for this function
So here we use the Gizduino UNO-SE stack on the bottom Layer

Align the A4 and A5 pin as this is where the I2C would be communicated



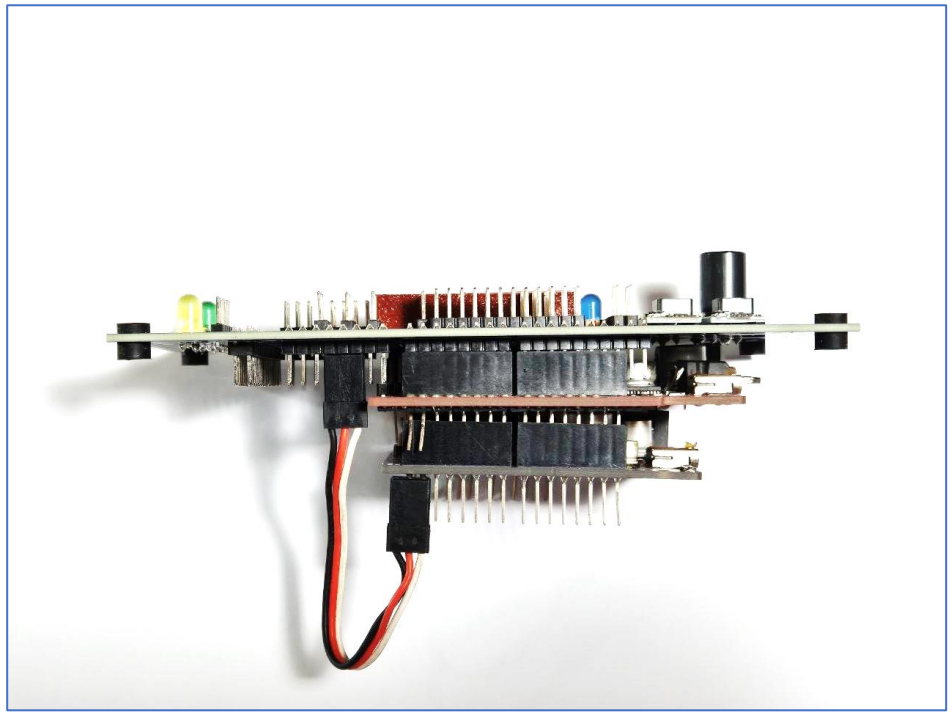


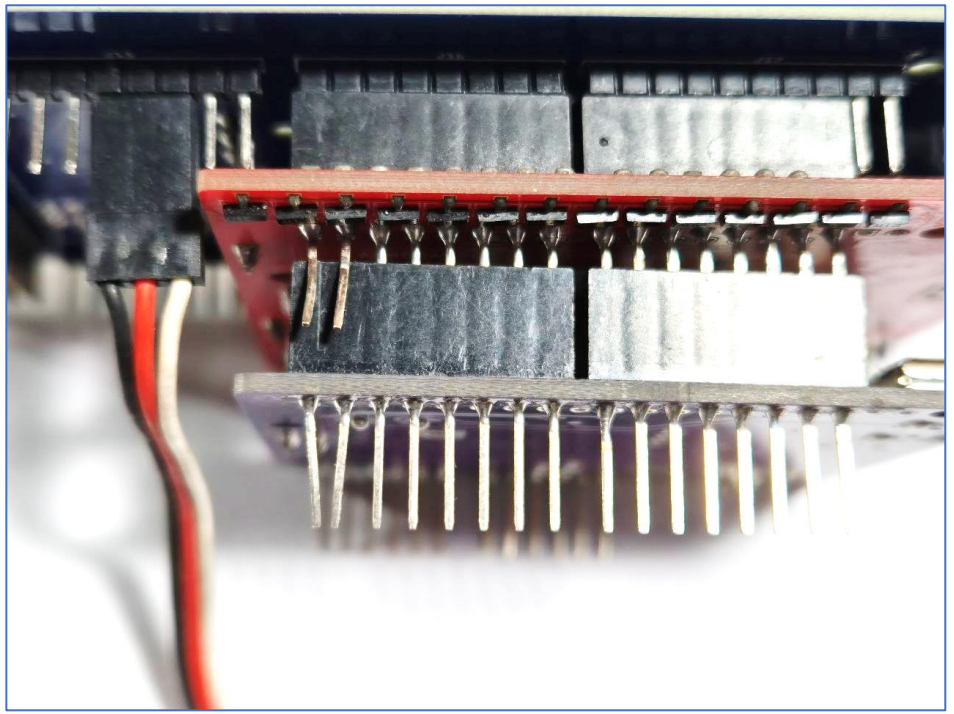
For the Serial notice the Gizduino Lin (Red Board) has its Serial Pin off TX and RX not inserted to the Gizduino UNO-SE (Purple board)

This is because we would be using the Serial pins of the Gizduino UNO-SE to communicate to the GPS

On the Right picture we bend the Serial pins TX RX to add friction . Otherwise Prefer its soldered in

On the Left Picture we get an RC Servo cable and hook it up TX2 (Synerduino) – TX0 (UNO-SE) and RX2 (Synerduino) - RX0 (UNO SE)



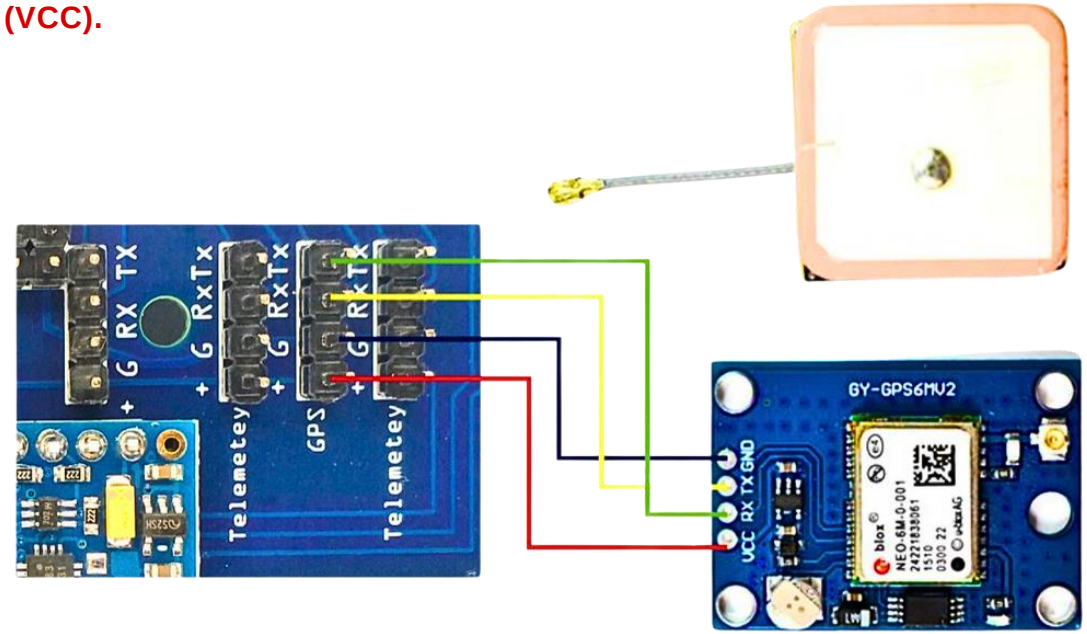


GPS CONFIGURATION

Finally GPS is hook up accordingly in this diagram

THE **TELEMETRY MODULE** INCLUDES PINS LABELED RX, TX, GND, AND VCC, WHICH CORRESPOND TO THE GPS MODULE'S MATCHING PINS.

TO CONNECT THESE COMPONENTS, COLOR-CODED WIRES CLARIFY THESE CONNECTIONS, WITH **YELLOW** INDICATING **RX TO TX**, **GREEN** FOR **TX TO RX**, **BLACK** FOR **GROUND (GND)**, AND **RED** FOR **POWER (VCC).**

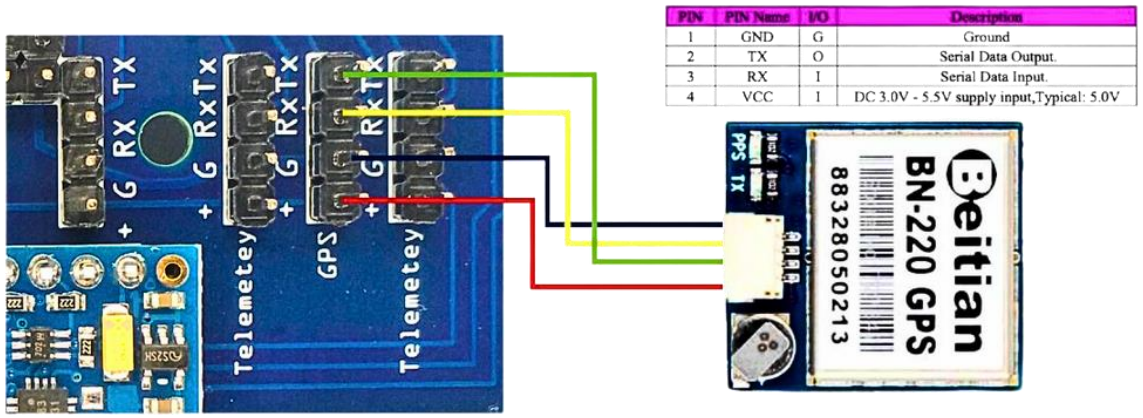


THE **FPV GPS MODULE** SHOWS A SPECIFIC PINOUT TABLE THAT DESCRIBES EACH PIN'S FUNCTION:

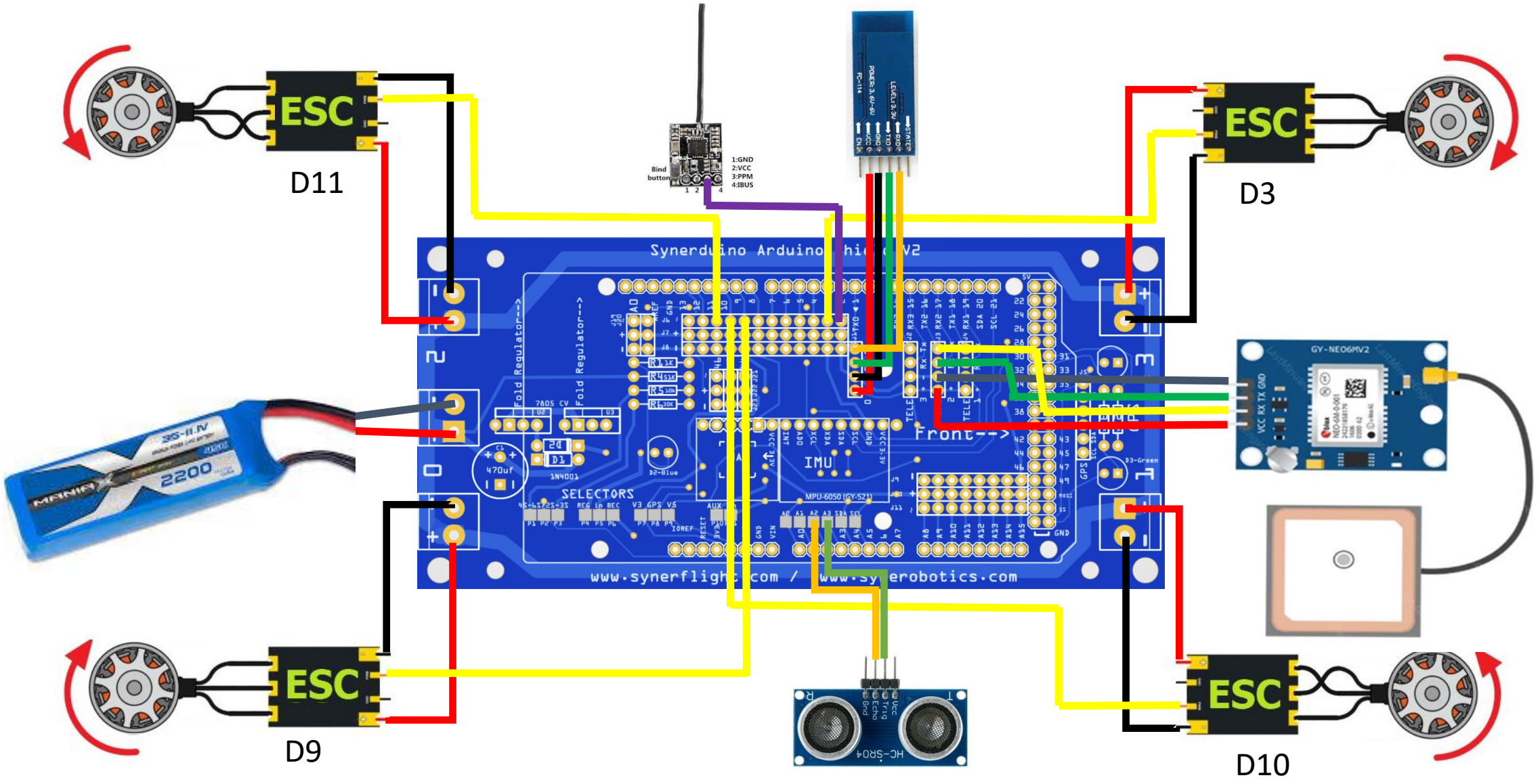
PIN 1 IS **GND (GROUND)**
PIN 2 IS **TX** (DATA OUTPUT)
PIN 3 IS **RX** (DATA INPUT)
PIN 4 IS **VCC** (POWER SUPPLY RANGES FROM 3.3V TO 5.0V)



PIN	PIN Name	I/O	Description
1	GND	G	Ground
2	TX	O	Serial Data Output.
3	RX	I	Serial Data Input.
4	VCC	I	DC 3.0V - 5.5V supply input,Typical: 5.0V



BATTERY, MOTOR, ESC, & PROPELLER INSTALLATION



Thank You

For more information , you can check the main documentation Tab or the Link to multirotor

https://synerflight.com/synerduino-documentation/1-getting-started/